基于时间序列算法的多模式切换算法

金怀智[1]，张英[2]，陈伟[2]，王顺利[1]，袁峰[1]

（1. 哈尔滨工业大学仪器科学与工程学院，哈尔滨，150001；

2. 北京航天自动控制研究所，北京 100070）

摘要：芯片温度的上升将会导致电子设备寿命大大降低，故障无法及时发现使得可靠性降低。常用的解决办法是通过休眠使得芯片在空闲期间尽快降温，为了不影响设备使用，这些功能会被安排在系统空闲时运行。这些功能的运行伴随着的一些问题，如在休眠时的瞬间启动将造成系统功率较大波动使得电源模块寿命下降，在空闲期间进行故障诊断可能影响设备启动。因此，如何合理的利用空闲时间，如何在系统空闲期间合理的安排休眠以及故障诊断等模式成为了重要研究课题。本文针对反复工作的数据采集系统进行模式管理开展研究，工作模式包括正常工作模式、浅度休眠模式、深度休眠模式和自检模式。针对空闲时间长度的不同，设计了不同的模式管理方案。此外，设计了用于评估模式管理效果的指标，该指标为有效管理率，有效管理率为系统正确选择模式管理方案同时顺利执行对应模式的概率。基于空闲时间预测算法来实现模式管理，根据预测的空闲时间长度来选择合适的模式管理方案，对模式管理策略进行了测试，针对实验结果对模式管理策略进行了改进，通过将不同阈值和权值进行组合进行测试，找到了最佳的阈值和权值的组合使得有效管理率达到了90.34%。

关键词：多模式切换算法、时间序列算法、预测算法、FPGA数据采集系统

中图分类号：TN406

Time series algorithm based multi mode switching method

JIN Huaizhi[1], ZHANG Ying[2], CHEN Wei[2],WANG Shunli[1],YUAN Feng[1]

(1. School of Instrumentation Science and Engineering, Harbin Institute of Technology, Harbin 150001;

2. Beijing Institute of Aerospace Automatic Controls, Beijing 100070)

Abstract: The rise in chip temperature will greatly reduce the lifespan of electronic devices, and the inability to detect faults in a timely manner will lead to a decrease in reliability. The commonly used solution is to cool down the chip as soon as possible during idle periods through sleep mode. In order not to affect device usage, these functions are scheduled to run when the system is idle. The operation of these functions is accompanied by some problems, such as the sudden start-up during sleep, which will cause significant fluctuations in system power and reduce the lifespan of the power module. Fault diagnosis during idle periods may affect device start-up. Therefore, how to make reasonable use of idle time, and how to arrange sleep and fault diagnosis modes during system idle periods have become important research topics. This article conducts research on pattern management for data acquisition systems that work repeatedly. The research modes include normal working mode, shallow sleep mode, deep sleep mode, and self check mode. Different mode management schemes have been designed for different lengths of idle time. In addition, an indicator was designed to evaluate the effectiveness of pattern management, which is the effective management rate. The effective management rate is the probability that the system correctly selects the pattern management scheme and successfully executes the corresponding pattern. Based on the idle time prediction algorithm, pattern management is implemented. The appropriate pattern management scheme is selected according to the predicted idle time length. The pattern management strategy is tested and improved according to the experimental results. By combining different thresholds and weights for testing, the optimal combination of thresholds and weights is found, resulting in an effective management rate of 90.34%.

作者简介：金怀智（1992-），男，博士研究生，主要研究方向：故障诊断、多模式研究、图像处理

通信联系人：袁峰（1963-），男，教授博导，主要研究方向：故障诊断、动态扭矩测量、视觉测量. E-mail: yuanf@hit.edu.cn

Key words: Multi-mode switching method; Time series algorithm; Prediction algorithm; FPGA data acquisition system

0 引言

在电子设备的设计过程中，如何提高可靠性和延长寿命是一个重要课题。由于设备运行过程中会散发的热量使得温度升高，温度升高使得芯片失效的概率提升，所以需要在不影响正常工作的前提下降低功耗，常用的也是比较有效的办法就是设置休眠模式，通过休眠来辅助降温[1]。为了不影响设备正常工作，休眠模式一般在设备空闲时启动。休眠模式下，设备内部功能会被适当的关闭，当处于深度休眠时通常只有主控的小部分功能在运行，当需要启动设备的正常工作程序时，为了保证一定的启动速度，会出现设备内部模块短时间内几乎同时启动的情况，这样会引起设备功率较大的波动，反而会给电源模块带来较大的负担，使得电源模块的寿命降低[2]。但是，如果不让系统处于深度休眠，只让少部分模块处于休眠就无法充分缓解系统的发热，所以如何管理休眠模式成为了一大关键。

系统功耗由多个方面组成，每种功耗的形成原因和影响程度也不同，其中影响最多的因素是动态功耗、漏电功耗以及短路功耗[3]。随着制造方法的改进，集成电路的复杂度得以增加，随之而来的系统的功耗也在随之升高，进而影响了电路系统的可靠性、同时还增加了辅助散热装置的成本[4]。根据相关研究，芯片温度每上升 10℃，会造成芯片的可靠性降低一半[5]。在导弹系统中，低功耗是导弹技术发展的趋势，也是弹载控制系统要注意的重要问题[6]。

目前，很多企业和高校投入大量资源开展低功耗器件研究。型号为 MSP432TM[7]的平台采用的是 1.8 V 到 3.6 V 的电源电压以减小功率值，经过检验该 ARM 平台可以做到在 RAM 数据持续保存的状态下耗损的电流仅 0.18μA，其休眠模式的电流仅 850 nA，做到了性能足够和降低功耗两方面的要求。浙江大学的杨艺丹[8]等人研制出一种功耗高电源抑制比亚阈值基准电压源，功耗最低可达 446.2 nW。中国科学院微电子研究所的胡安俊[9]等人设计了一种超级功耗的全 CMOS 基准电路，该电路通过亚阈值设计方法来降低电路的功耗，使得最低能提供 1 nA 的基准电流时也能提供 560 mV 的基准电压。

此外，现在针对功耗控制的方案很多。比如根据各个功能分区的特点，设计与之相适应的低功耗方案，可以采用多中电源电压或者多种阈值，根据各个模块的特点设计阈值电压来控制亚阈值电流进而减小静态功率耗损[10]。使用 CMOS 门控时钟技术和门控电源技术来控制电路各个模块的电源和时钟，来减少功率损耗[11]。除了设计电路来减少功耗以外，还可以添加功耗管理策略，比如睡眠模式下关闭不必要的功能和模块来降低整体功耗[12]。

本文的研究的主要内容是不影响系统正常工作的情况下，合理调度设备的各个模式，以达到降低功耗，延长寿命，提高可靠性的要求。为了延长寿命，需要使用低功耗技术，使得在深度休眠模式下尽可能的降低功耗，深度休眠模式下部分芯片的睡眠功能将被开启，部分电源电压将会降低，高频时钟切换成低频时钟来降低动态功耗，只将模式切换的控制和响应单元保持在运行状态，系统进入最低功耗状态，以减小系统发热，使得系统温度尽快降至合理水平。为了减小系统的所有芯片同时启动时功率波动对电源模块造成的负担，设计了浅度休眠模式，该模式用于深度休眠模式与正常工作模式之间的过渡。为了保证可靠性，还设计了故障诊断功能，该功能在自检模式下运行，自检模式下，标准信号会输入到相应模块，通过检测模块输出来判断被测模块是否故障。本文根据空闲时间的长度划分了不同的模式管理状态，模式管理单元需要通过预测空闲时间长度来选择不同的模式管理方案。具体研究内容

如下：（1）设计并搭建基于 FPGA 为核心的硬件平台，可对各个芯片的使能端进行控制，部分芯片的电源通过功率开关控制，这部分功能用于深度休眠模式和浅度休眠模式；（2）建立模式管理模型。研究深度休眠模式、浅度休眠模式、自检模式以及正常工作模式的实现；（3）研究空闲时间预测算法，并将合适的空闲时间预测算法加入到模式管理策略中，根据所选择的空闲时间预测算法的特点，设计本文的模式管理策略；（4）研究模式管理策略的 FPGA 实现。

1 实验方法

1.1 FPGA 数据采集系统平台设计

模式管理策略是在 FPGA 中运行，平台的主控单元选择的是具有低功耗功能的赛灵思 Kintex-7 系列 FPGA。由于多通道数据采集系统普遍存在于各类设备当中，在设备中起到了重要作用，所以正常工作模式设定为多通道数据采集和一些数据通信。

研究的工作模式有正常工作模式，浅度休眠模式，深度休眠模式和自检模式。正常工作模式设定为多通道数据采集和数据通信，数据采集功能开启时 FPGA 将会通过数据采集通道采集外部传感器数据以及硬件平台内部参数，数据通信是 RS-232 通信。浅度休眠模式和深度休眠模式是控制各个芯片的休眠功能，FPGA 通过控制功率控制执行器来启动和关闭各个芯片的休眠功能。自检模式下 FPGA 通过控制故障诊断执行器来实现故障排查。系统总体方案如图 1 所示。图 2 为搭建的硬件平台实物图。

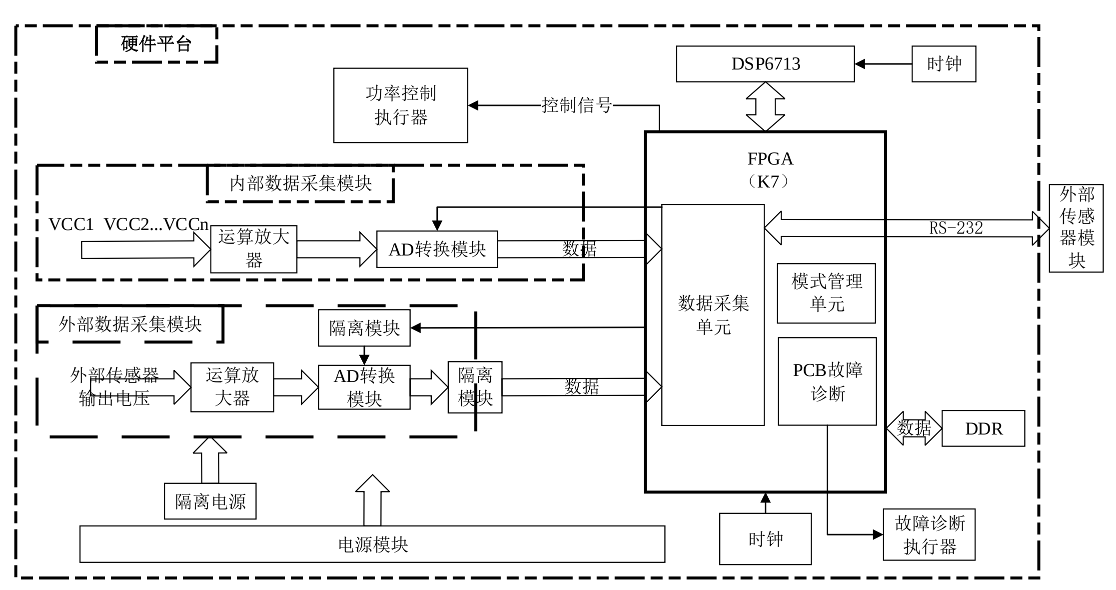


图 1 系统总体方案图

Fig. 1 System overall design drawing

其中，数据采集系统的硬件平台主要包括 Kintex-7 系列 FPGA、TMS320C6713、数据采集模块、外部通信接口、功率控制模块以及故障诊断模块。该硬件平台的电源设计具有多种特点。电压类型多，部分电源需要较大功率，同时有模拟电源和数字电源，同时为提高可靠性还需要设置隔离电源。

FPGA 和 DSP 电压类型很多。为了在休眠模式下尽可能的降低功耗，FPGA 将选择 Kintex-7 系列的-2LE（低功耗扩展型）芯片，该芯片的内核电压 VCCINT 有两种，分别是

1.0 V 和 0.9 V，低功耗模式下,静态功耗将会下降 25%到 26%。其中比较特殊的 LTM4677 被用于输出 1.0 V 或者 0.9 V 电源。通过切换电源模块的设置电阻来切换电压。LTM4633 的 3.3 V 电源为 DSP 供电，LTM4671 的 3.3V 电源为 FPGA 供电。

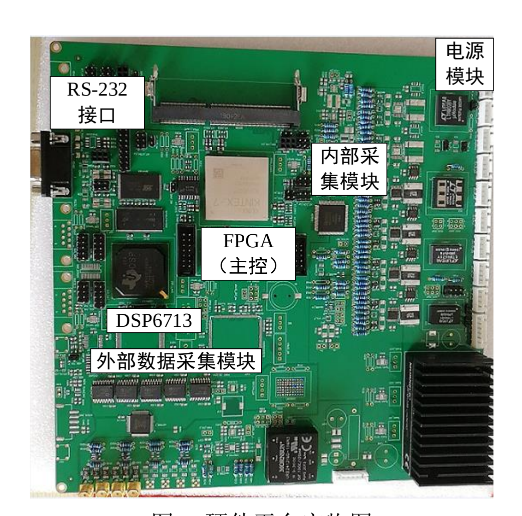


图 2 硬件平台实物图
Fig. 2 Hardware platform

功率较大的电源芯片有 LTM4671、LTM4633、LTM4649 以及 LTM4677，均自带使能端，软启动功能。由于开关电源产生的噪声较大，所以在模拟电路的电源设计中，常采用线性电源。线性电源的特点是噪声小，输入输出电压压差越大，效率越低。为了保证线性电源的效率，首先使用开关电源将电压转换至合适电压，然后使用 LDO 将电源转换成所需的模拟电源电压。

考虑到 FPGA 和 DSP 需要一定的上电顺序，将电源模块输出 12 V 电源以后首先上电的是 LTM4677、LTM4633 和 LTM4671 的 1.2 V 部分以及隔离部分电源，隔离部分上电以后处于待使能状态，等待系统完成上电以后启动隔离部分芯片功能。LTM4677、LTM4633、LTM4649 以及 LTM4671 的每个电源通道都有 RUN 引脚和 PGOOD 引脚。RUN 引脚的阈值均在 1.1 V 到 1.3 V 之间，输入到 RUN 引脚的电压超过阈值时，电源模块的对应通道就能输出电压。PGOOD 的引脚均为开漏输出，将电源稳定输出电压以后，PGODD 可通过上拉电阻输出高电平。

主控芯片相关设计：通过对比多款 FPGA，选择了赛灵思的 Kintex-7 系列 FPGA 作为主控芯片。Kintex-7 系列 FPGA 采用 28nm 节点实现最佳成本/性能/功耗的平衡，高达 478K 逻辑单元，支持高达 32 路 12.5G 收发器、2.845GMAC、34Mb 的 BRAM，Kintex-7 系列 FPGA 与相似密度的 40nm 器件相比，功耗降低了 50%，同时还具有低功耗功能，十分适合用于本文硬件平台的主控。

时钟设计：为了适应各种模式以及各个功能，FPGA 有多个晶振，有正常模式下的时钟，高速传输的时钟，DDR 通讯时的时钟以及低功耗模式下的时钟。一共选型了 4 款时钟，包括：LTC6991HS6#TRMPBF、TT32L327K50ITR、DSC1001BL5-100.0000、DSC1101BI5-133.0000。

DSP 设计：因为本文模式管理研究的平台是数据采集系统，通常数据采集系统都带有数据分析功能，所以平台上有 DSP 模块，为平台提供传感器数据分析能力。正常工作模式下，DSP 通过 EMIF 总线与 FPGA 进行通讯。深度休眠模式下，DSP 启动自带的低功耗模式。浅度休眠模式下，DSP 退出低功耗模式，数据总线和地址总线处于高阻态，等待使能

信号，接收到使能信号以后进入正常工作状态，开始与 FPGA 通信。DSP 选型为 TMS320C6713B，这款 DSP 有 32 位 EMIF 总线，一个 32 位定时器，264KB 的 RAM，主频可高达 300MHz。当工作在 225MHz 时，运行速度可达 1350MFLOPS，1800MIPS，使用双固定/浮点乘法器，可使运行速度高达 450MMACS，满足本硬件平台要求。DSP 将为加载到硬件平台的算法提供数据处理功能。

数据采集模块设计：硬件平台的数据采集模块分为两个部分，一个用于测量硬件平台的电流、电压和温度，属于内部数据采集系统，另外一个用于测量外部传感器，属于外部数据采集系统。

数据采集系统：内部数据采集系统用于采集硬件平台的电流、电压和温度。外部数据采集系统用于采集外部传感器的模拟信号。外部数据采集系统一共四个通道，每个通道会接入一个传感器，一个备用传感器和一个用于故障诊断的固定电压信号。为了保证可靠性，外接的传感器会留有备用传感器，当外部传感器故障时，会通过模拟开关将备用传感器信号接入。为了对数据采集通道的故障情况进行判断，每个通道将会由模拟开关接入固定电压，当 AD 转换模块输出的数值异常时判定为故障。

1.2 模式管理软件设计

研究的模式有正常工作模式、浅度休眠模式、深度休眠模式和自检模式（表 1）。正常工作模式下，FPGA 通过数据采集模块获取传感器的模拟信号，通过 RS-232 接口获取外部传感器的数据，为了对传感器数据进行信号处理，FPGA 将数据通过 EMIF 总线发送给 DSP，DSP 将处理的数据发送给 FPGA，至此，数据采集系统完成了一个周期的正常工作。深度休眠模式下，整个系统将只有 FPGA 的部分程序处于运行状态。浅度休眠模式，整个系统处于就位状态。自检模式，对关键模块进行故障诊断。本章将探讨以上模式以及模式切换的程序流程。

表 1 各模块工作状态

Tab. 1 Working condition of the modules

工作模式	FPGA	DSP	数据采集模块	通讯芯片
正常工作模式	正常工作	正常工作	正常工作	正常工作
浅度休眠模式	待使能	就位	就位	就位
深度休眠模式	待激活	休眠	休眠	休眠
自检模式	部分工作	休眠	部分工作	部分工作

每种工作模式都有最短工作时间，当空闲时间有限时需要根据各个工作模式的最短工作时间来对工作模式进行选择。根据空闲时间的长度进行划分，系统空闲时的工作模式以及模式切换方案各有不同。方案分类如图 3 所示：

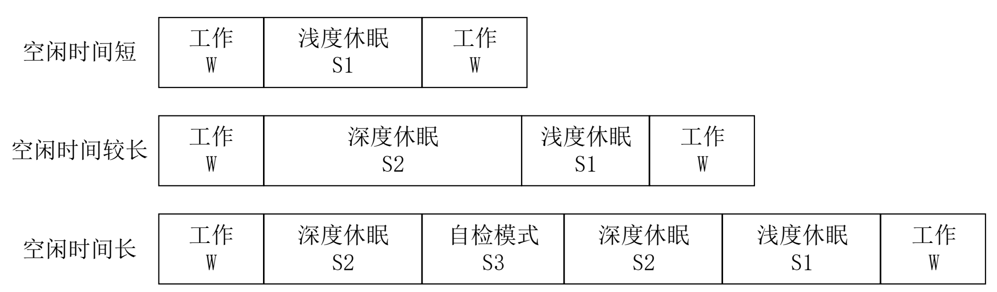


图 3 工作模式方案分类

Fig. 3 Classification of the working mode schemes

（1）正常工作模式设计：DSP 的 EMIF 总线通讯速度最高大约为 4MB/s，由于 EMIF 总线同时连接 FLASH 和 SDRAM，所以将正常工作模式下 FPGA 通过 EMIF 与 DSP 通讯最大速度调整为 1MB/s。由于内部数据采集模块采集硬件平台的电压、电流和温度等参数，这部分数据仅用于监控，所以 AD7616 的 16 个通道的采样率设定为 100sps。由于外部数据采集通道是采集外部传感器数据，同时考虑到 FPGA 与 DSP 的通讯最大速度为 1MB/s，所以将 AD7606 的采样率设定为 10Ksps。RS-232 通讯的波特率设定为 11500bps。正常工作模式以 AD7616 完成一次采样为周期。正常工作模式程序流程如下：

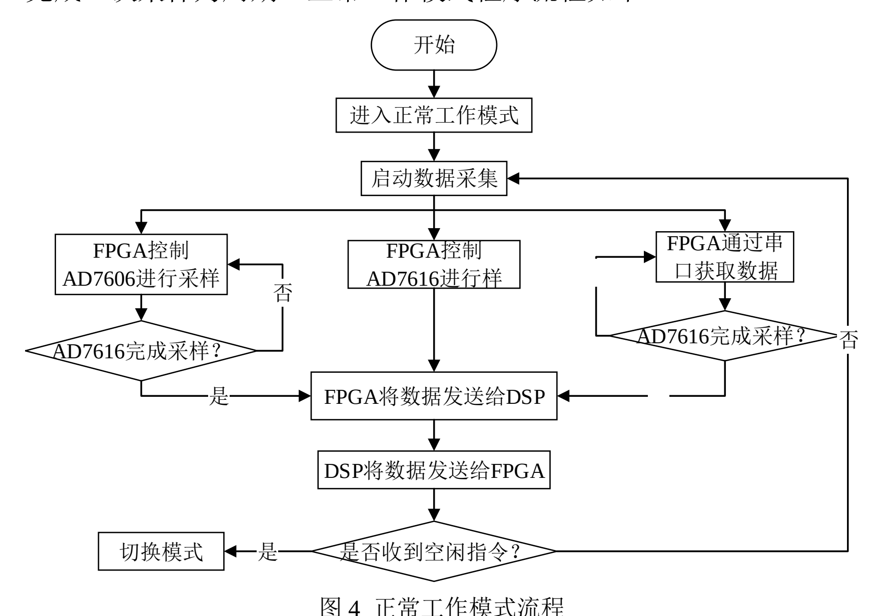


图 4 正常工作模式流程

Fig. 4 Normal working mode process

（2）休眠模式软件设计：休眠模式分成了浅度休眠模式和深度休眠模式。不仅要考虑模式状态对功耗的影响，同时还需要考虑休眠模式切换成正产工作模式的时间不能超过 1 秒。首先对需要进行各个芯片的低功耗模式进行研究。需要进行休眠控制的芯片有 AD7606、AD7616、模拟电路电源、RS-232 通讯芯片、DSP 以及 FPGA。浅度休眠模式的目标保证系统处于就位状态，FPGA 收到模式切换指令以后能快速控制系统进入正常工作模式。所以对各个芯片的模式安排如下：

表 2 浅度休眠模式各个芯片的状态

Tab. 2 Status of each chip in the shallow sleep mode

芯片	状态
AD7606	待机模式
AD7616	正常模式，FPGA 不给转换信号
模拟电路电源	正常模式
RS-232 通讯芯片	自动掉电模式
DSP	正常状态，所有程序暂停
FPGA	内核电压升为 1V，门控时钟功能开启

对浅度休眠模式切成正常工作模式过程的时间进行分析。浅度休眠模式下，首先需要对 AD7606 和 RS-232 通讯芯片进行唤醒，两个芯片可以同时进行唤醒，同时唤醒消耗的时间为 100 μs，然后 FPGA 开始进行工作，加上模式切换的时间大约为 200 μs，满足模式切换速度的需求。

深度休眠模式的目标是最大程度降低系统功耗，但同时保证接收到启动指令以后，切换

成正常工作模式的时间不超过 1 秒。深度休眠模式下需要考虑两个因素。第一是深度休眠模式下 FPGA 必须能运行响应启动指令的程序，并且当预测的空闲时间足够长，FPGA 需要能执行自检模式控制程序。第二是启动时间比较长的是 FPGA 和 DSP，如果深度休眠模式下要将系统处于最低功耗状态，启动时间可能超过 1 s。针对第一个因素的解决方法是深度休眠模式下将 FPGA 切换成 0.9V 内核电压状态，而不是复位状态，仅内核电压降低时 FPGA 仍能执行程序，而复位状态下 FPGA 不能执行程序。针对第二个因素，DSP 功耗最低的状态是复位状态，把 DSP 从复位状态唤醒需要较长时间，FPGA 从内核电压 0.9 V 切换成 1.0 V 大约需要 200 ms，DSP 从复位状态切换成正常状态需要 250 ms，考虑到上电顺序的需要，并且 DSP 兼具控制 FPGA 内核电压切换的任务，所以模式切换时间可能超过 1 秒。为此综合以上两点，深度休眠模式下，DSP 设定为 PD1 模式，FPGA 设定为内核电压为 0.9 V。

表 3 深度休眠模式下各芯片状态

Tab. 3 Status of each chip in the deep sleep mode

芯片	状态
AD7606	关断模式
AD7616	关断模式
模拟电路电源	不启动
RS-232 通讯芯片	掉电模式
DSP	PD1 模式
FPGA	内核电压 0.9 V，使用频率最低的时钟，停止部分程序

（3）自检模式软件设计：本文研究的对象是数据采集系统模式管理，所以核心硬件是数据采集，所以自检模式下，将会对内部数据采集系统和外部数据采集系统进行故障诊断。自检模式下，FPGA 已经在深度休眠模式时切换成内核电压为 0.9 V 的状态，此时的 FPGA 仍能执行程序。

内部数据采集包括电流测量、电压测量和温度测量。由于电压测量功能是针对系统的电源电压，这部分电压已知，只需判断是否超过正常范围即可。电流的测量针对系统个部分的电源，波动较大。电流检测电路进行故障诊断时，双通道模拟开关 MAX20327 会断开电流采样电阻两端连线，然后将固定电压输入电路中，然后 FPGA 控制 AD7616 执行一次模数转换，检查 AD7616 的输出的数据是否超过正常范围。温度检测的故障诊断是通过对比同一温度测量点的三个温度传感器，当常用传感器与另外两个传感器数据不一致时判定该传感器故障。

外部数据采集系统用于对外接的传感器的模拟信号进行采样，进行故障诊断时，可通过单通道模拟开关 ADG602 断开传感器信号，然后通过第三个模拟开关接入固定电压，然后 FPGA 通过控制 AD7606 完成一次模数转换，检查 AD7606 输出是否异常来判断外部数据采集系统是否故障。

为了尽量将空闲时间用于休眠，自检模式应该尽量快速，同时因为 FPGA 具有并行处理能力，所以内部和外部数据采集系统的故障诊断同时进行。自检模式程序流程如下：

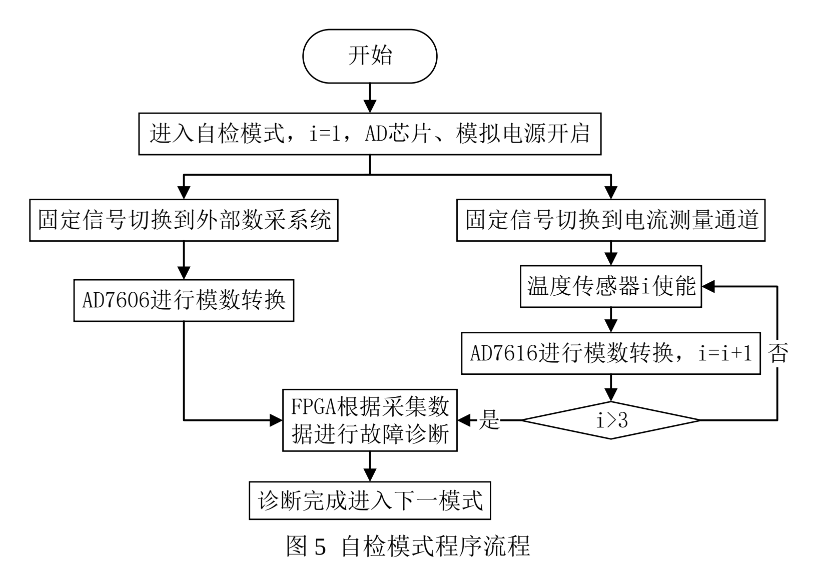


图 5 自检模式程序流程

Fig. 5 Program flow of the self-check mode

（4）模式切换软件设计：由于空闲时间预测算法精度不可能完全准确，切换成正常工作模式的外部指令在任何时刻都有可能出现，在突发情况发生时，启动指令可能出现在预测的空闲时间段的任意时刻，为了应对突发情况，系统对所有情况都做模式切换预案。可能的模式切换有以下几种情况：

i. 正常工作模式与浅度休眠模式的相互切换：正常工作模式与浅度休眠模式的相互切换最为简单，也最为快速。就是 FPGA 通过门控时钟的方法，控制所有功能的时钟驱动的开启和关闭，同时 FPGA 通过 IO 改变 AD7606 和 RS-232 芯片的模式即可。相互切换的时间为 200 μs。

ii. 深度休眠模式与浅度休眠模式的相互切换：在这类情况下，系统每个部分都要切换状态。在这类情况下都是先对 FPGA 的状态进行切换，首先完成 FPGA 内核电压的切换，FPGA 内核电压从 1.0 V 切换成 0.9 V 或者 0.9 V 切换成 1.0 V，当浅度休眠转为深度休眠时，FPGA 给 DSP 发送模式切换指令，DSP 执行切换成 PD1 模式的程序完成模式切换。当浅度休眠转为深度休眠时，FPGA 通过 IO 口给 DSP 发送中断信号，使得 DSP 退出 PD1 模式。FPGA 在控制 DSP 进行模式切换的同时，也同时控制其他芯片进行模式切换。所以这类情况下，相互切换的时间为 450 ms。

iii. 深度休眠模式和自检模式的相互切换：深度休眠模式下，FPGA 仍能执行程序，所以深度休眠模式切换成自检模式时，FPGA 直接运行自检模式控制程序即可。在深度休眠模式切换成自检模式的过程中，首先将 AD7606、AD7616 从关断状态切换成工作模式，同时将模拟电路电源开启，然后完成对数据采集系统的故障诊断，完成故障诊断以后，再将 AD7616 和 AD7606 切换成关断模式，系统再次切换回深度休眠模式。系统在此类模式切换花费的时间等于 AD7616 和 AD7606 的芯片模式切换的时间，最大为 30 ms。

iv. 自检模式切换成正常工作模式：自检模式切换成正常工作模式属于突发情况，该情况可能发生在自检模式的任意阶段。自检模式任意阶段的运行，都不影响 FPGA 内核电压的切换，由于切换成正常工作模式的优先级最高，所以自检模式的数据允许丢弃并且工作状态也允许被打断。所以此类情况，直接按照深度休眠模式切换成正常工作模式的流程切换模式。

v. 深度休眠模式与正常工作模式相互切换：深度休眠下首先进行的 FPGA 内核电压的切换，然后 FPGA 控制 DSP、AD 芯片、模拟电路电源以及 RS-232 通讯芯片的状态切换。

1.3 基于预测算法的模式管理策略

现在比较常用的时间序列预测算法有指数平滑算法、ARIMA 算法和 GM(1,1)算法等。因为本文将采用指数平滑算法，因为本部分将对该预测算法进行介绍。

设备使用的时间序列通常蕴含着一定的规律性，这种规律通常和使用环境和使用者本身有关系，所以在使用环境以及使用者不变时，在符合一定条件下，时间序列的规律通常变化不大，这意味可以通过历史数据对未来数据进行预测。指数平滑算法就是通过这一原理来预测时间序列。指数平滑算法是根据前一时刻的预测数据和真实数据对下一时间的时间进行预测，通过权重参数来决定当前预测数据对历史预测数据和历史实际数据的依赖程度。指数平滑算法的公式如下：

$$T_c(k) = a*T(k-1) + (1-a)*T_c(k-1) \tag{1}$$

式中，$T_c(k-1)$——前一次的时间预测值；$T_c(k)$——当前的时间预测值；$T(k-1)$——前一次的时间实际值；a——权重参数。

如果将历史预测结果带入公式（4-3），将会得到以下结果：

$$T_c(k) = a*T(k-1) + a*(1-a)*T(k-2) + \cdots + a(1-a)^{k-1}T(0) + (1-a)^{k-1}T_c(0) \tag{2}$$

在指数平滑算法中，权重参数a表示实际数据与预测数据的比例，所以权重参数a的取值范围为$0 \le a \le 1$，当随着预测次数越来越多时，预测的结果近似的等于实际时间序列的加权平均，数据越靠近当前次数，在预测中所占的比重也越大，所以指数平滑预测算法相当于通过局部的实际时间序列来进行预测，而这个预测过程依靠单一参数a来控制。为了更好的分析该预测算法，首先取a等于 0.8，输入 100 个时间序列，实际数据序列采用随机函数加上正弦函数生成，将实际数据序列与预测数据序列进行对比，获取仿真图如图 6 所示：

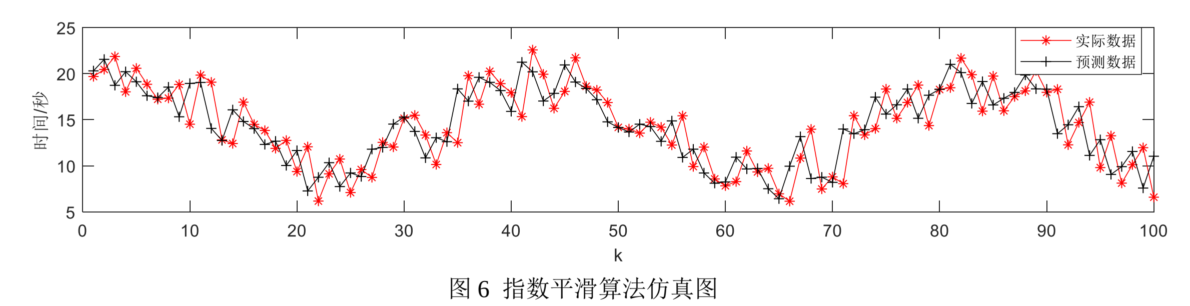


图 6 指数平滑算法仿真图

Fig. 6 Simulation diagram of exponential smoothing algorithm

可以看出指数平滑预测算法能跟随实际数据的趋势进行预测，为了检验算法精度，采用相对均方误差进行衡量。公式如下：

$$MRE = \frac{\sum_{k=1}^{N}\left(\frac{T(k)-T_c(k)}{T(k)}\right)^2}{N} \tag{3}$$

式中，MRE——为相对均方误差；N——时间序列长度。

图 6 中参数a 取值为 0.8，MRE 为 0.1696，误差较大，由于预测精度和参数有很大关系，为了研究最佳参数，设计一个重复次数为 100 次的仿真实验。仿真总共重复进行了 100 次，仿真时权重参数a取值为 0.05 到 0.95。该仿真实验的目的是通过 100 次的重复实验，找到最佳参数a，观察每次 MRE 最小的参数是否不变。实验结果如图 7 所示：

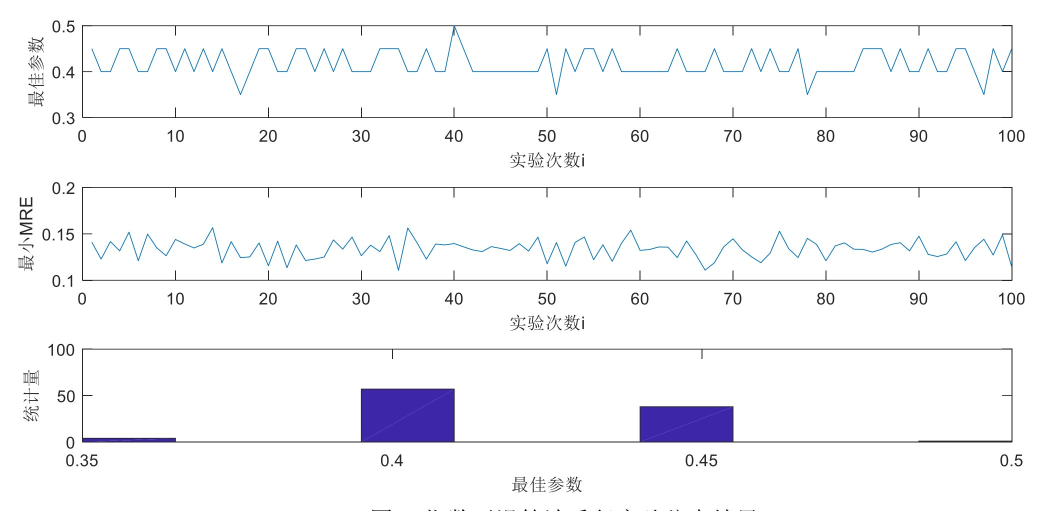


图 7 指数平滑算法重复实验仿真结果

Fig. 7 Repetitive experimental simulation results of exponential smoothing algorithm

在仿真过程中，MRE 最小的参数 *a* 为最佳参数。通过重复实验可以看出，最佳参数在 0.35 到 0.5 之间，取参数 *a* 为 0.4，获得仿真图如下：

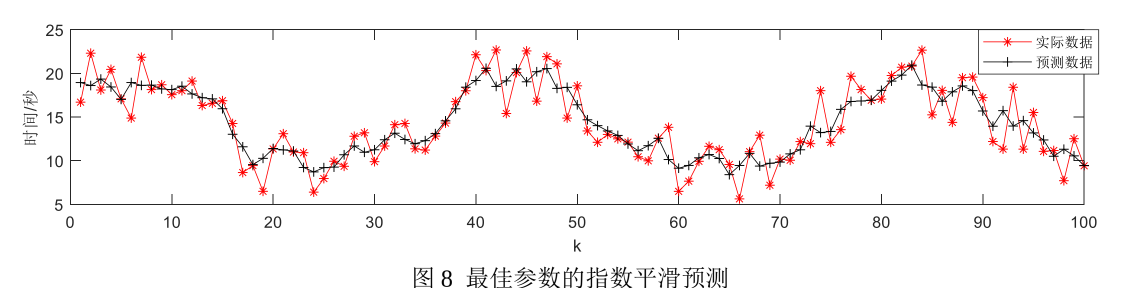


图 8 最佳参数的指数平滑预测

Fig. 8 Exponential smoothing prediction of optimal parameters

在最佳参数下，指数平滑算法预测结果的 MRE 值为 0.1284。

1.4 模式管理测试方案设计

深度休眠模式或者自检模式切换成正常工作模式时，芯片几乎同时切换状态可能给电源模块造成一定的负担，从而使得电源模块的寿命降低，为此最好的情况是系统先进入浅度休眠模式再切换成正产工作模式，系统切换成浅度休眠模式时各模块是依次切换状态，引起的电源输出功率波动小，同时浅度休眠下系统各模块个几乎都处于就位状态，启动时不会造成较大功率波动。本文要求在任何情况下系统都能在 1 秒内进入正常工作模式，即本文要求的第一位是快速启动，由于环境以及重复性问题，系统的各个模式切换状态的时间会有波动，所以收到启动时间不立即切换成正常工作模式可能影响设备运行，为此收到启动信号时，无论系统当前处于什么模式，都必须直接切换成正常工作模式。总结来说，有效的模式管理是收到启动信号时系统正好处于浅度休眠模式。

为了方便实验分析，对模式管理方案和模式进行编码。根据空闲时间的长短，有三种模式管理方案。模式管理方案 1，空闲时间短，系统空闲期间只运行浅度休眠模式。模式管理方案 2，空闲时间较长，系统在空闲期间先运行深度休眠模式，再运行浅度休眠模式。模式管理方案 3，空闲时间长，系统在空闲期间依次运行深度休眠模式、自检模式、深度休眠模式、浅度休眠模式。实际情况中，还存在空闲时间很短的情况，该情况下，空闲期间不切换模式，该情况定义为模式管理方案 0。模式的编号以及模式管理方案选择表如下：

表 4 模式代码符号表

Tab. 4 Pattern code symbol table

模式	深度休眠模式	浅度休眠模式	自检模式
模式代号	1	2	3
模式符号	S1	S2	S3

表 5 模式管理方案选择表

Tab. 5 Mode Management Scheme Selection Table

空闲时间/秒	模式管理方案	方案代号	依次运行的模式
小于阈值 1	模式管理方案 0	1	无
阈值 1 到阈值 2	模式管理方案 1	2	S2
阈值 2 到阈值 3	模式管理方案 2	3	S1、S2
大于阈值 3	模式管理方案 3	4	S1、S3、S1、S2

为了更好研究模式管理的效果，本文定义了两种指标，分别是模式管理误差和有效管理率。模式管理误差分成两项，一项是模式管理方案误差，另一项是模式切换误差。

模式管理方案误差定义为以真实空闲时间对应的模式管理方案代号为真值，把系统通过预测的空闲时间选择的模式管理方案代号为样本值，模式管理方案误差等于样本值减去真值。公式如下：

$$EPMS = \alpha - \beta \tag{4}$$

式中，*EPMS*——为模式管理方案误差；α——为样本值，即空闲时间预测数据对应模式管理方案代号；β——为真值，即空闲时间实际数据对应模式管理方案代号。

模式切换误差针对具体模式。模式切换误差定义为以实际空闲时间对应的模式管理方案下应该运行的模式数量为真值，把接收到启动信号时已经运行过的模式数量作为样本值，模式切换误差等于样本值减去真值。

$$EMS = m_\alpha - m_\beta \tag{5}$$

式中，*EMS*—为模式切换误差；m_α—为样本值，即空闲期间实际已经运行过的模式数量；m_β—为真值，即应该运行的模式数量。

表 5 列出了空闲时间长度对应的模式管理方案和对应的模式。为了评估本文研究的模式管理的效果，引入指标有效管理率η，有效管理率η定义为在空闲期间，系统选择对应的最佳模式管理方案并将对应模式全部运行的概率。有效管理率η定义如下：

$$\eta = \frac{n}{N} * 100\% \tag{6}$$

式中，*n*——为 *EPMS* 和 *EMS* 同时等于 0 的实验次数；*N*——为总实验次数。

该概率通过重复实验测出，在实施多次实验过程中，每次实验输入不同长度的空闲时间，FPGA 会根据预测的空闲时间选择模式管理方案和依次运行的模式，每次实验结束后 FPGA 会将该次实验的模式管理方案代号和运行的模式代号输出，重复实验结束后，首先求出实际空闲时间序列对应的模式管理方案代号和模式代号，并以此为真值，然后统计 FPGA 输出结果中与真值一致的次数，最后计算出有效管理率η。

图 9 中的 N 为实验重复的次数，i 大于 1 以后模式管理策略将生效。收到系统空闲信号以后，系统根据预测的空闲时间长度选择合适的模式管理方案，然后将模式管理方案的代号输出到 txt 文本中，然后系统根据选定的模式管理方案依次运行模式，在收到启动信号以后，

FPGA 自动将当前模式代号输出到 txt 文本中，同时系统根据当前模式选择对应的模式切换程序。为了方便获取模式管理方案代号和模式代号，测试将在 modelsim 中进行，为了加快仿真速度，输入的空闲时间缩小 1024 倍，相应的 0.01 秒为周期的时钟 clkms 提升为 102.4KHz 时钟，输入的时间序列的阶码将加上 10。

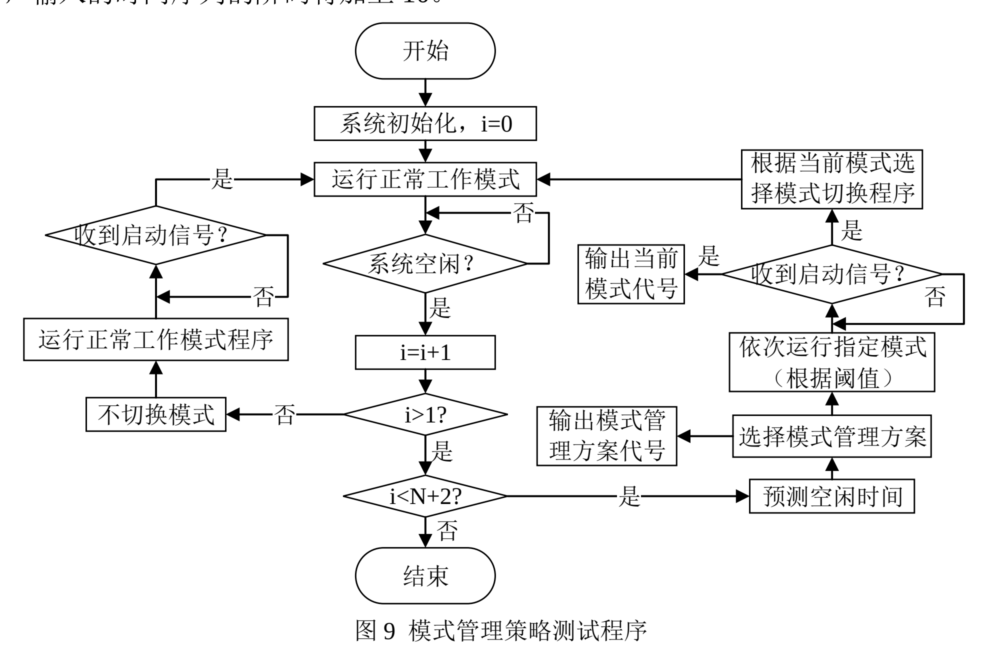


图 9 模式管理策略测试程序
Fig. 9 Pattern management strategy testing program

2 实验与结果分析

本文设计了三个步骤的实验。步骤一初步测试各个阈值下模式管理策略的效果，以缩小最合适的阈值范围，同时对实现结果进行分析以提出优化建议。步骤二对优化思路进行测试，同时确定所有阈值的最佳取值。步骤三详细分析最终确定的模式管理策略的测试结果。

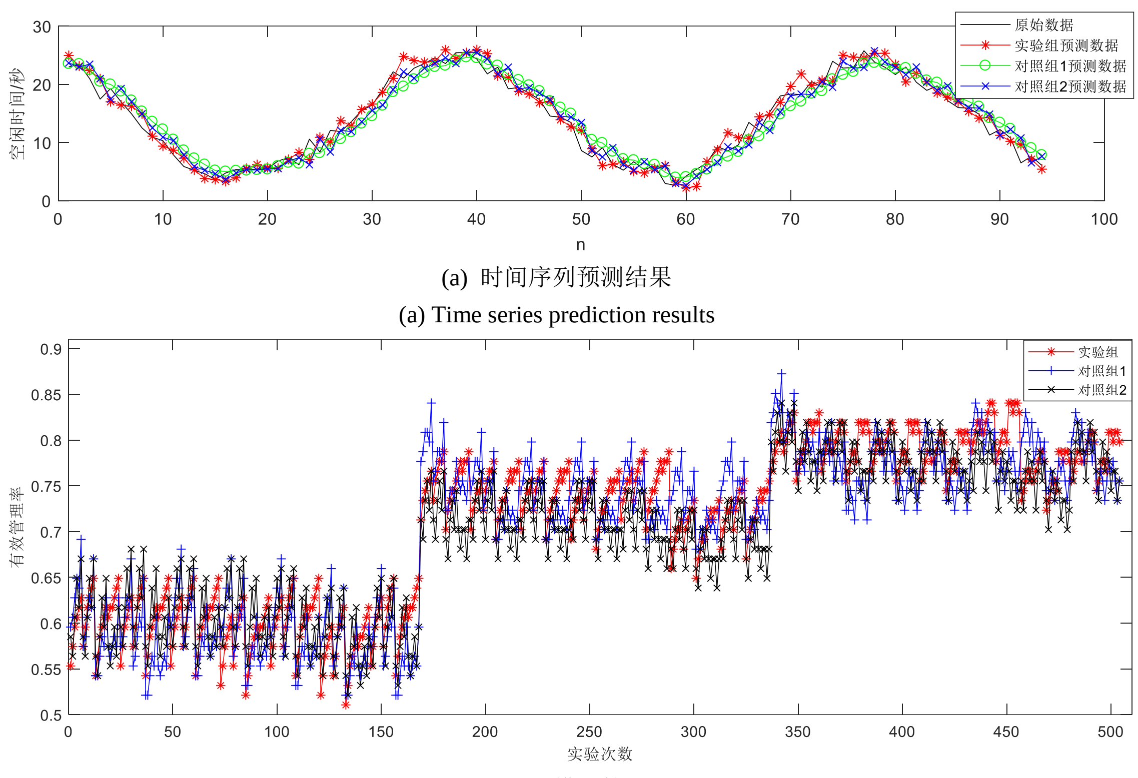
2.1 步骤一

首先通过实验确定模式管理中的阈值。阈值 1 范围为 4 到 7，阈值 2 范围为 8 到 13，阈值 3 范围为 14 到 20，模式阈值 1 的范围为 1 到 3。在步骤一中，实验组和对照组的设置如下：

表 6 步骤一：实验组和对照组设置情况
Tab. 6. Step 1: Setting up experimental and control groups

项目	采用的预测算法
实验组	前 5 次用指数平滑算法，之后用优化的 GM(1, 1)算法
对照组 1	使用指数平滑算法
对照组 2	采用上一时刻实际数据作为本次空闲时间预测值

实验时阈值的变化梯度为 1，用于测试的阈值数据预先转化为 IEEE-754 格式存储在 txt 文本中，然后依次将阈值输入到 FPGA 程序中，每次实验输入的时间序列含有 100 个数据。由于实验组前 5 次使用的指数平滑算法，和对照组 1 一摸一样，无对比意义，所以测试结果只分析了后 95 次实验的结果。



(a) 时间序列预测结果

(a) Time series prediction results

(b) 不同阈值下的模式管理策略测试

(b) Mode management strategy testing under different thresholds

图 10 步骤一：模式管理策略测试试验结果

Fig. 10 Step 1: Test results of mode management strategy

测试时空闲时间序列为同一空闲时间序列，实验对阈值的所有组合进行了测试，一共 504 种组合，所以一共 504 次实验。从图 10(b)中可以看出大约在第 342 次左右对照组 1 的有效管理率 η 最高，但是实验组未达到最大值。第 450 次左右实验组的有效管理率 η 达到最大值，但是对照组 1 和对照组 2 的有效率管理率 η 并未达到最大值。为了更详细的观察数据，将第 340 到 344 次实验以及第 441 到 456 次实验的数据列入下表中.

表 7 第 340 次到 344 次以及第 441 到 456 次实验数据

Tab. 7 Experimental data from the 340*th* to 344*th* and 441*st* to 456*th* sessions

实验次数	阈值 1/秒	阈值 2/秒	阈值 3/秒	模式阈值 1/秒	有效管理率 η		
					实验组	对照组 1	对照组 2
340	4	11	14	3	0.7979	0.8404	0.8298
341	4	12	14	3	0.7979	0.8298	0.7979
342	4	13	14	3	0.8085	0.8723	0.8404
343	5	8	14	3	0.7872	0.7979	0.7979
344	5	9	14	3	0.7979	0.8085	0.7660
441	5	10	18	3	0.8298	0.8191	0.7872
442	5	11	18	3	0.8404	0.8085	0.8085
443	5	12	18	3	0.8404	0.7872	0.7766
444	5	13	18	3	0.8298	0.8085	0.7979
445	6	8	18	3	0.7766	0.7553	0.7553
446	6	9	18	3	0.7872	0.7553	0.7234
447	6	10	18	3	0.7979	0.7766	0.7660
448	6	11	18	3	0.8085	0.7660	0.7872
449	6	12	18	3	0.8085	0.7447	0.7553

450	6	13	18	3	0.7979	0.7660	0.7766
451	7	8	18	3	0.8404	0.7872	0.7660
452	7	9	18	3	0.8404	0.7766	0.7340
453	7	10	18	3	0.8298	0.7872	0.7553
454	7	11	18	3	0.8404	0.7660	0.7553
455	7	12	18	3	0.8404	0.7447	0.7234
456	7	13	18	3	0.8298	0.7660	0.7447

从表 7 可以看出，在步骤一有效率管理率 η 的全局最大值为 0.8723， 在第 342 次实验的对照组 1 中出现。实验组的有效管理率 η 的最大值为 0.8404，在第 442、443、451、452、454 以及 455 次实验出现。同时，可以得出一个结论，模式阈值 1 可取值为 3。其他阈值需要进一步做实验确定。为了进一步分析，获得预测数据残差图如下：

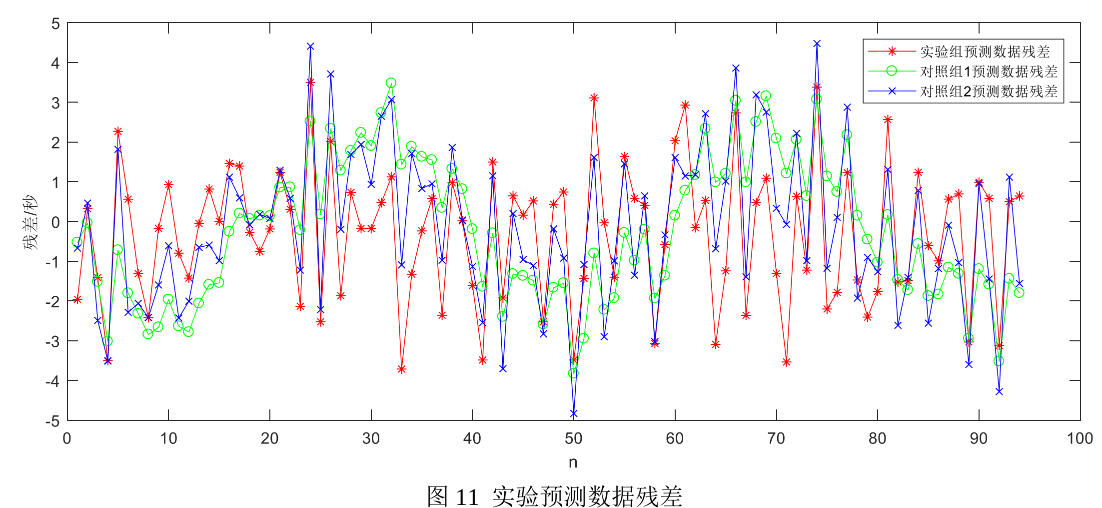


图 11 实验预测数据残差

Fig. 11 Experimental prediction data residual

实验组、对照组和对照组 2 的 *MRE* 分别是 0.1336、0.1438 和 0.1499。实验组的预测精度最高，但是有效管理率 η 的全局最大值却出现在对照组 1，说明提高有效管理不仅需要关注预测精度，还需要关注算法预测结果的特点。实验组残差的波动大，近似于白噪声波形。对照组 1 的残差的波动小，有跟随输入数据整体浮动的趋势。实验组的预测算法 GM(1, 1) 算法具有反应快的特点，对照组 1 的预测算法具有残差局部波动相对较小的特点。当空闲时间变化较快时，实验组的效果更好，当空闲时间处于阈值边界时残差波动小的对照组 1 效果更好。本文将采用将两者进行加权平均的方法进行改进，公式如下：

$$Tm = km * Tg + (1 - \mathrm{km}) * Te \qquad (7)$$

式中，Tm —为改进模式管理策略的空闲时间预测值；Tg —为优化的 GM(1, 1)算法的空闲时间预测值；Te —为指数平滑算法的空闲时间预测值；km —为加权系数，范围为 0 到 1。

2.2 步骤二

根据步骤一的测试和分析，空闲时间预测算法改进为指数平滑算分预测结果与优化的 GM(1,1)算法预测结果的加权平均。为了更好测试效果，将实验组的预测算法换成改进的预测算法。对照组 1 依然使用指数平滑算法进行预测，由于步骤一中对照组 2 的效果明显差一些，所以对照组 2 的预测算法换成优化的 GM(1,1)算法。步骤二，实验组和对照组设置如下：

表 8 步骤二：实验组和对照组设置情况

Tab. 8 Step 2: Setting up experimental and control groups

项目	时间预测算法
实验组	优化的 GM(1, 1)算法和指数平滑算法的加权平均
对照组 1	指数平滑算法
对照组 2	优化的 GM(1, 1)算法

实验组和对照组 2 前五次依然都采用指数平滑算法进行预测。本步骤实验的目的有两个。第一是找到最佳权值使得实验组的有效管理率 η 最大。第二是通过实验结果的分析，找到阈值 1、阈值 2 和阈值 3 的最佳值。

为了达到以上目的，将会对权值 km 、阈值 1、阈值 2 以及阈值 3 进行组合，然后将每个组合都进行测试，测试其对有效管理率 η 的影响。km 取值从 0.1 到 0.9，取值梯度为 0.1，一共 9 个值。阈值 1 取值 4 到 7，取值梯度为 1，一共 4 个值。阈值 2 取值 8 到 13，取值梯度为 1，一共 6 个值。阈值 3 取值为 14 到 20，取值梯度为 1，一共 7 个值。将会有 1512 种组合。将一共进行 1512 次实验，实验结果如下：

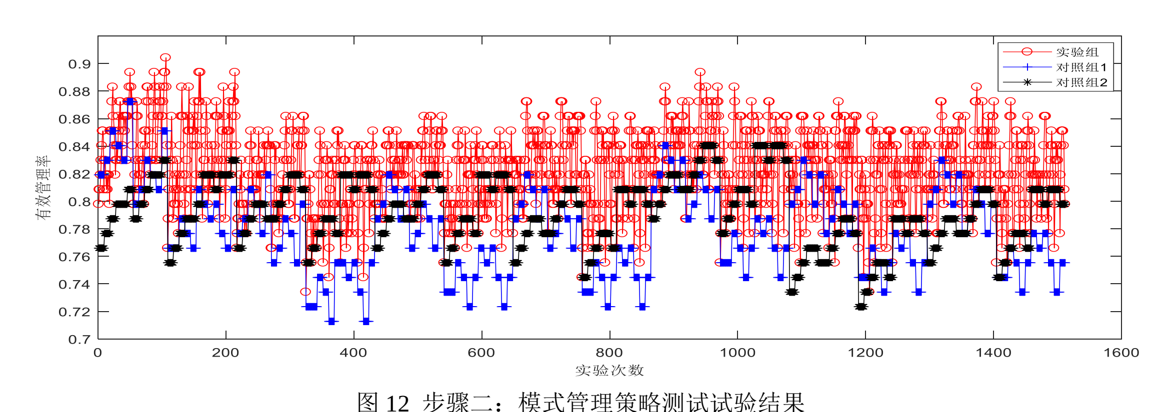


图 12 步骤二：模式管理策略测试试验结果

Fig. 12 Step 2: Test results of mode management strategy

从图中可以看出，全局最大值大约在 106 次实验。将 104 到 108 次实验数据列表如下：

表 9 第 104 次到 108 次实验数据

Tab. 9 Experiment data from104th to 108th

实验次数	阈值/秒			权值 km	有效管理率 η		
	阈值 1	阈值 2	阈值 3		实验组	对照组 1	对照组 2
104	5	13	14	0.5	0.8936	0.8511	0.8298
105	5	13	14	0.6	0.8936	0.8511	0.8298
106	5	13	14	0.7	0.9043	0.8511	0.8298
107	5	13	14	0.8	0.8830	0.8511	0.8298
108	5	13	14	0.9	0.8404	0.8511	08298

从上表可以看出，实验组的有效管理率 η 达到了 0.9043，达到了本文模式管理目标，所以将第 106 次实验的参数定为本文最终的参数。

经过一系列实验测试本文模式管理过程中各个阈值和权值的取值如下：

表 10 模式管理策略阈值与权值最终取值

Tab. 10 Threshold and weight values for pattern management strategy

项目	阈值/秒					权值 km
	阈值 1	阈值 2	阈值 3	模式阈值 1	模式阈值 2	
值	5	13	14	3	12	0.7

2.3 步骤三

经过前两个步骤，完成了对模式管理策略的改进，以及最佳阈值的选择，使得有效管理率η超过了 90%。接下来将对确定参数以后的模式管理策略进行详细分析。为了取得更好对比效果，步骤三增加一个对照组，对照组 3 的预测数据为前一时间的实际数据。

表 11 步骤三：实验组和对照组设置情况

Tab. 11 Step 3: Setting up experimental and control groups

项目	预测算法
实验组	优化的 GM(1, 1)算法和指数平滑算法的加权平均，权值km为 0.7
对照组 1	指数平滑算法
对照组 2	优化的 GM(1, 1)算法
对照组 3	采用上一时刻实际数据作为本次空闲时间预测值

本步骤对同一阈值不同预测算法的模式管理策略进行测试，对有效管理率η、空闲时间预测精度、深度休眠次数、深度休眠总时间长度、浅度休眠次数、浅度休眠总时间长度以及自检模式执行次数进行了统计。统计结果如下：

表 12 步骤三：实验统计数据

Tab. 12 Step 3: Experimental Statistical Data

项目	实验组	对照组 1	对照组 2	对照组 3
有效管理率η	0.9043	0.8511	0.8298	0.8404
空闲时间预测精度（*MRE* 值）	0.1153	0.1438	0.1336	0.1499
深度休眠次数	52	52	52	52
深度休眠总时长/秒	835.3	819.0	845.0	825.1
浅度休眠次数	84	87	79	83
浅度休眠总时长/秒	414.3	295.0	214.8	279.6
自检模式执行次数	48	47	47	47

从表中可以看出以下结论：（1）实验组的有效管理率最高，该数值代表了实验组在模型方案选择和模式切换执行方面表现最佳；（2）并不是空闲时间精度越高，有效管理率η就越高。对照组 1 的预测精度就比对照组 2 高，但是有效管理率比对照组 2 低。当权值km等于 0.6 时，实验组的空闲时间预测精度达到了 0.1141，但是有效管理η只有 0.8936，实验组将指数平滑算法和优化的 GM(1,1)算法进行结合，合理利用了两种算法各个的特点，使得管理效果得到提升；（3）有效管理率最高但是深度休眠时间不一定最长，但是对电源模块造成的负担更小。实验组的深度休眠时间略低于对照组 2，但是实验组深度休眠次数多于对照组 2，这代表对照组 2 与实验组相比，直接从深度休眠模式切换成正常工作模式的次数更多。如果空闲前期系统处于深度休眠模式，为了减小对电源模块的负担，当空闲时间即将结束时，系统需要进入浅度休眠模式，所以实验组与对照组 2 相比，在空闲时间即将结束时及时切换成浅度休眠模式的次数更多，所以实验组深度休眠的时间稍短，但是这样对电源模块造成的负担更小，更符合本文模式管理的目标；（4）有效管理率高则进入自检模式的次数可能更多。实验组比对照组多 1 次进入自检模式，这样就使得实验组能比对照组更加及时的发现故障；（5）有效管理率高则总休眠时间能更长。虽然实验组的深度休眠时间不是最长的，但是浅度休眠时间远远大于其他组，这表示实验组能比其他组获得更加充分的休眠以使得系统的热量散发的更快，温度处于合理范围的概率更大，寿命更长。

3 结论

本文将常用的多通道数据采集系统设定为本文的研究平台，同时基于课题需要，对具有功耗控制和故障诊断功能的多通道数据系统进行了硬件平台设计、模式设计、模式切换设计、预测算法研究和实现以及模式管理策略的设计和实现。本文的主要研究成果如下：

设计了系统的工作模式，包括正常工作模式、浅度休眠模式、深度休眠模式和自检模式。对模式管理模型和预测算法进行了研究分析。通过对模式管理模型的分析，确定了不同情况下的模式管理方案以及相关阈值的范围。选择指数平滑算法和 GM(1,1)作为本文的空闲时间预测算法。根据这两种算法预测所需的数据量以及模式管理模型设计了模式管理策略，模式管理策略是通过对空闲时间进行预测，然后根据预测的空闲时间合理安排浅度休眠模式、深度休眠模式和自检模式在空闲期间的运行和切换。

设计了验证模式管理的指标和实验，并完成了本文模式管理策略的测试。设计的指标为有效管理率，该指标代表了系统能够正确选择模式管理方案并完成模式的执行的概率，通过重复实验确定了最佳阈值，通过对预测算法进行改进使本文得模式管理略的有效管理率达到了 90.34%，达到了本文的目标。对最佳的模式管理策略进行了详细实验分析，本文的模式管理策略能让系统在空闲期间处于休眠的总时间最长同时保证对电源模块的负担更小，从而使得系统寿命延长，提高完成故障诊断的次数使得系统及时发现故障的概率增加。

本文虽然对模式管理进行了设计、研究和测试，但是还是存在一些不足，总结如下：

（1）在 FPGA 中实现模式管理策略时没有太注意程序的资源占用率以及运行速度。未来研究对程序进行资源和速度的优化，使得功耗和延时减小，从而减小模式管理策略本身的消耗带来的影响。

（2）预测算法的精度会随着数据规律的变化而变化，模式管理策略的效果也会之波动，需要进行自适应方面的研究。

[参考文献] (References)

[1] Xiao X , Guo J , Chang C ,et al.Mode Switching Technique for High Efficiency Buck Converter[J].Journal of circuits, systems and computers, 2023(9):32.DOI:10.1142/S0218126623501591.

[2] Nakamura T , Saito S , Fujimoto K ,et al.Time-Division Multiplexing for FPGA Considering CNN Model Switch Time[C]//2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW).IEEE, 2021.DOI:10.1109/IPDPSW52791.2021.00074.

[3] Kaenel V V , Macken P , Degrauwe M G R .A Voltage Reduction Technique for Battery-Operated Systems[J].IEEE Journal of Solid-State Circuits, 1990, 25(5):1136-1140.DOI:10.1109/4.62134.

[4] Shuya K , Hiroshi T , Mary I .Low power consumption FPGA accelerator with real time current measurement system[J].Ipsj Sig Notes, 2014, 2014:1-6.

[5] Mehta D S , Mishra V , Verma Y K ,et al.A Hardware Minimized Gated Clock Multiple Output Low Power Linear Feedback Shift Register[J]. 2020.DOI:10.1007/978-981-32-9775-3_33.

[6] Stanger I , Shavit N , Taco R ,et al.Robust Dual Mode Pass Logic (DMPL) for Energy Efficiency and High Performance[J].IEEE, 2020.DOI:10.1109/ISCAS45731.2020.9181127.

[7] 德州仪器半导体技术有限公司. 德州仪器推出32位MSP432(TM)微控制器[J]. 电脑与电信，2015(3): 8-10.

[8] 杨艺丹,赵梦恋,刘胜,董阳涛,吴晓波.一种低功耗高电源抑制比亚阈值基准电压源[J].微电子学与计算机,2018,35(08):109-112+116.

[9] Kumar B N M , Rangaraju H G .Design and implementation of high-speed and low-power consumption Moore-based loopback adder on FPGA[J].International Journal of Intelligent Unmanned Systems, 2021, ahead-of-print(ahead-of-print).DOI:10.1108/IJIUS-09-2020-0056.

[10] 胡安俊,胡晓宇.一种超低功耗全CMOS基准电路[J].电子设计工程,2019,27(01):14-18+24.

[11] Andrejevic M , Litovski V , Zwolinski M .Fault Diagnosis in Digital Part of Mixed-Mode Circuit[C]//International Conference on Microelectronics.IEEE, 2006.DOI:10.1109/ICMEL.2006.1650986.

[12] Chen L , Gong C , Li C ,et al.Low Power Convolutional Architectures: Three Operator Switching Systems Based on Forgetting Memristor Bridge[J].Sustainable Cities and Society, 2021, 69:102849.DOI:10.1016/j.scs.2021.102849.